



دانشگاه صنعتی شریف

دانشکده مهندسی برق

آزمایشگاه پیشرفته برنامه‌نویسی

"برنامه‌نویسی موازی و Multi-Threading"

آزمایش ششم

مقدمه

در این آزمایشگاه می‌خواهیم با برنامه‌نویسی Multi-Thread آشنا شویم.

Thread Loop -1

ابتدا برنامه‌ای بنویسید که یک thread بسازد که thread دیگری ایجاد می‌کند. Thread جدید خود یک thread می‌سازد و به همین ترتیب هر thread، thread جدیدی می‌سازد و این روند ادامه پیدا می‌کند تا 50 thread ساخته شود. هریک از این thread ها "Hello from Thread <num>" را چاپ می‌کند. برنامه باید جوری باشد که threadها به صورت برعکس سلام کنند یعنی ابتدا thread 50 ام سلام کند سپس thread 49 ام و ...

Race Condition -2

در خصوص مفهوم Race Condition تحقیق کرده و نحوه استفاده از آن در ایجاد آسیب‌پذیری‌های پرداخت (Payment Vulnerability) شرح دهید. آیا چندبار استفاده از یک کد تخفیف، از جمله موارد این آسیب‌پذیری است؟

فرض کنید کلاس Counter به شکل زیر تعریف شده باشد.

```
1
2 public class Counter {
3     static int count = 0;
4
5     static void inc() {
6         if(count <= 100) {
7             try {
8                 Thread.sleep(10);
9             } catch (InterruptedException e) {
10                e.printStackTrace();
11            }
12
13            count++;
14        }
15
16        if(count > 100) {
17            System.out.println("Failed");
18        }
19    }
20 }
21 }
```

برنامه‌ای بنویسید که 100 thread ایجاد کند که به صورت همزمان تابع inc() را صدا می‌کنند. آیا به صورت مفهومی امکان دارد عبارت "Failed" در خروجی چاپ شود؟ چرا؟

اگر تعریف inc() را به static synchronized inc() تغییر دهیم، چه اتفاقی می‌افتد؟

Merge Sort -3

برنامه ای بنویسید که آرایه ای 256 عضوی به عنوان ورودی بگیرد و آن را به صورت نزولی با استفاده از الگوریتم Merge Sort مرتب کند. مرتب سازی ادغامی را به صورت bottom up پیاده سازی کنید و در هر مرحله، merge قسمت های مختلف باید به صورت همزمان و در thread های مختلف انجام شود.

مثال زیر نحوه کارکرد bottom up merge sort را برای آرایه 8 تایی $A[1:8]$ نشان می دهد.

$A = [1, 7, 4, 3, 8, 6, 2, 5]$

ابتدا زوج های $[1, 7]$ ، $[4, 3]$ ، $[8, 6]$ و $[2, 5]$ طی فرایند merge مرتب می شوند:

$A = [1, 7, 3, 4, 6, 8, 2, 5]$

سپس $[1, 7]$ و $[4, 3]$ ، مرج می شوند و مرتب شده در آرایه قرار می گیرند. به همین شکل در انتهای این مرحله، $[2, 5, 6, 8]$ مرتب شده در آرایه قرار می گیرد:

$A = [1, 3, 4, 7, 2, 5, 6, 8]$

و در نهایت با مرج شدن $A[1:4]$ و $A[5:8]$ ، A مرتب خواهد شد:

$A = [1, 2, 3, 4, 5, 6, 7, 8]$